

# Optimal Stochastic Convex Optimization Through The Lens Of Active Learning

Aaditya Ramdas and Aarti Singh  
Machine Learning Department, Carnegie Mellon University  
{aramdas, aarti}@cs.cmu.edu

## Abstract

The large fields of convex optimization and active learning have been developed fairly independent of each other, from the design of algorithms to the techniques of proof. Given the growing literature in both these subjects, we believe that understanding the connections between them is important to people in both areas. Here, we establish few such interesting relationships in upper and lower bound techniques that bring out these similarities. Our prime result is showing upper and lower bounds for precisely how the minimax rate for optimizing a given function depends solely on a flatness/noise condition for the function around its minimum.

## 1 Introduction

Almost all convex optimization algorithms are, by design, of a sequential nature, with future steps depending on the results of past actions. This gives them a very natural flavour found in active learning, which deals with sequential sampling strategies with the aim of minimizing a loss function. One can naturally ask if there is much in common between these two fields, given the natural similarity in stating their objectives. In this paper, we answer the above question in the strongly affirmative, by connecting concepts found in both. Furthermore, we demonstrate algorithms and proof techniques from one field to solve problems in the other. The central problem we deal with is stochastic convex optimization, as introduced in the next section.

We generalise the well-known concept of  $\kappa$ -Uniform Convexity ( $\kappa$ -UC), to a weaker notion of Generalised  $\kappa$ -Uniform Convexity ( $\kappa$ -GUC) that only describes the flatness of a function at its optimum. We prove lower bounds for how fast we can optimize such functions using methods from active learning, and show that these bounds are indeed achieved by a recent variant of gradient descent. This work implies a strong result, that a convex function's behaviour around its minimum is the only factor that the minimax rates for its optimization depend on, and that the function estimation error looks like  $\Theta(T^{-\frac{\kappa}{2\kappa-2}})$  after  $T$  steps. While UC only allows  $\kappa \geq 2$ , GUC allows  $\kappa > 1$  which yields rates much faster than  $O(1/T)$  for those functions. We also prove that the point estimation error, which is not often considered by the optimization community, looks like  $\Theta(T^{-\frac{1}{2\kappa-2}})$ .

Our work bears some similarity to [6] where the authors use techniques for convex optimization analysis to derive lower bounds for active learning in one dimension. However, they do not derive rates or give algorithms for convex optimization, and our results are broader and richer. Another work that bears a resemblance is [4], which derives upper bound rates for UC functions. Our upper bounds are achieved by tuning the recent Epoch-Gradient-Descent for  $\kappa > 1$  whereas [4] analyze primal-dual subgradient methods

for  $\kappa \geq 2$ . They also do not show lower bounds, or connections to active learning, and use the UC (as opposed to GUC). We get the same rates as they do of  $O(T^{-\frac{\kappa}{2\kappa-2}})$ , which vary smoothly between  $O(T^{-1})$  for strongly convex functions and  $O(T^{-1/2})$  for convex functions. However, UC does not permit  $1 < \kappa < 2$ , but GUC does, and for these, we can obtain rates that are faster than  $O(T^{-1})$ , which is quite surprising and interesting.

To begin with, we define the oracle model for stochastic convex optimization from the seminal [5] and the more recent [1], who proved tight lower bounds for convex and strongly convex classes. On introducing the Generalised- $\kappa$ -Uniform-Convexity, we point out its relationship to the Tsybakov Noise Condition (TNC) [7] that is popular in classification and level set estimation, and use this analogy to adapt an active learning algorithm to perform optimal minimization in one dimension. We show how to use active learning proof techniques from [2] to get minimax lower bounds for optimizing  $\kappa$ -GUC functions in any dimension. Finally, we get tight upper bounds by tuning the Epoch-GD algorithm [3] to achieve these rates in expectation and with high probability for all  $\kappa > 1$ .

## 2 Oracle Model of Stochastic Convex Optimization

Stochastic convex optimization in the oracle model can be defined as the task of minimizing a  $d$ -dimensional convex function  $f$  over a convex set  $S \in \mathbb{R}^d$ , when given oracle access to unbiased estimates of the function value and gradient at any point in  $S$ , by using as few queries to the oracle as possible. We follow the setup of [5] and [1], and summarise what is necessary for completeness.

A *first order oracle* is a (possibly random) function  $O : S \rightarrow \mathbb{R}^{d+1}$ , which answers a query  $x$ , by returning  $(\hat{f}(x), \hat{g}(x))$  such that  $\mathbb{E}[\hat{f}(x)] = f(x)$  and  $\mathbb{E}[\hat{g}(x)] = g(x)$ , where  $g(x) \in \partial f(x)$ . We additionally assume  $\hat{f}$  and  $\hat{g}$  have unit variance. Let the class of all such oracles be called  $\mathcal{O}$ .

An *optimization algorithm* is any procedure that solves the task of finding the optimum  $x_f^*$  by repeatedly querying the oracle at different points in  $S$ . The method can decide which points to query at based on the results of earlier queries, and tries to use as few queries as possible to achieve its task. We normally assume that it has no further knowledge of the functioning of the oracle. Define  $\mathcal{M}_T$  to be the set of methods that use  $T$  queries and finally return an estimated point  $\hat{x}_T$ .

The central question can be posed as follows: *How many queries will it take to get  $\epsilon$ -close to the optimal point?*, or equivalently as *How close can we get to the optimal point, given a budget of  $T$  queries (time-steps)?*. We use the second framework in this paper.

The *error bound* achieved by an algorithm can be measured in two ways, which we call point-distance and function-distance. For any  $M_T \in \mathcal{M}_T$  that returns  $\hat{x}_T$ , we define the *function-error* and *point-error* with respect to function  $f$  after  $T$  queries in  $S$  to oracle  $O$  respectively as:

$$\epsilon_T(M_T, f, S, O) = f(\hat{x}_T) - \min_{x \in S} f(x) = f(\hat{x}_T) - f(x_f^*)$$

$$\rho_T(M_T, f, S, O) = \text{dist}(\hat{x}_T, x_f^*) = \|\hat{x}_T - x_f^*\|_2$$

Given a class of functions  $\mathcal{F}$ , the *minimax error* is then defined as the expected error (over the randomness of the oracle) achieved by the game in which an adversary picks the oracle and the set, the learner picks an optimization procedure knowing  $S$  (but not the oracle's workings), and then the adversary picks a function that the learner must optimize. They can be defined formally as:

$$\rho_T^*(\mathcal{F}) = \sup_{O \in \mathcal{O}} \sup_{S \in \mathbb{R}^d} \inf_{M_T \in \mathcal{M}_T} \sup_{f \in \mathcal{F}} \mathbb{E}_O[\rho(M_T, f, S, O)] \quad (1)$$

$$\epsilon_T^*(\mathcal{F}) = \sup_{O \in \mathcal{O}} \sup_{S \in \mathbb{R}^d} \inf_{M_T \in \mathcal{M}_T} \sup_{f \in \mathcal{F}} \mathbb{E}_O[\epsilon(M_T, f, S, O)] \quad (2)$$

In this paper, we shall deal with  $\epsilon_T^*(\mathcal{F})$  and  $\rho_T^*(\mathcal{F})$  for different function classes  $\mathcal{F}$ , and will be interested in how it scales with  $T$ . The dependance on  $d$  may not be optimal, but that will not be the concern of this work - we will instead try to give a fine characterization in terms of their dependence on  $T$  and  $\kappa$ , a generalised uniform convexity parameter that will be introduced in the next section.

### 3 Relating Uniform Convexity (UC) and Tsybakov Noise Condition (TNC)

Given a closed, convex set  $S \in \mathbb{R}^d$ , let  $\mathcal{F}^C(S)$  be the set of all strictly convex functions on  $S$ , ie,

$$\forall x, y \in S, \quad \forall t \in [0, 1], \quad f(tx + (1-t)y) < tf(x) + (1-t)f(y)$$

We consider this restriction only because it implies that  $f$  has a unique minimum on any set  $S$ . We could alternately define it to be the class of all convex functions that have a unique minimum in  $S$ .

Let  $\mathcal{F}_{\lambda, \kappa}^{UC}(S)$  be the set of all  $(\lambda, \kappa)$ -uniformly convex functions on  $S$ .  $f$  is  $\kappa$ -UC on  $S$  with UC parameter  $\lambda$ , or  $f \in \mathcal{F}_{\lambda, \kappa}^{UC}(S)$  for  $\kappa \geq 2$  (necessary condition, Appendix), if we have

$$\forall x, y \in S, \quad \forall t \in [0, 1], \quad f(tx + (1-t)y) \leq tf(x) + (1-t)f(y) - \frac{1}{2}\lambda t(1-t)\|x - y\|_2^\kappa$$

$\kappa = 2$  is well known as the class of  $\lambda$ -strongly convex functions  $\mathcal{F}_\lambda^{SC}$ . As shown in [4], if a uniformly convex function  $f$  is subdifferentiable at  $x$ , then for any subgradient  $g_x \in \partial f(x)$ ,

$$f(y) \geq f(x) + g_x^\top(y - x) + \frac{\lambda}{2}\|x - y\|_2^\kappa$$

Taking  $x$  to be  $x_f^*$  and noting that  $0 \in \partial f(x_f^*)$ , we get

$$f(x) - f(x_f^*) \geq \frac{\lambda}{2}\|x - x_f^*\|_2^\kappa$$

Readers familiar with classification literature will recognize the similarity of this last expression to the Tsybakov Noise Condition (TNC) [7]. The TNC is often used in classifications problems (given  $x$ , predict its label  $\ell(x)$ ) to describe the flatness of the regression function  $\eta(x) = P(\ell(x) = 1|x)$  around the decision boundary  $\eta(x) = 1/2$ . For example, in one dimension, the following condition is assumed to hold in some region around the minimum, for some value of  $\kappa$ :

$$|\eta(x) - 1/2| \geq c|x - x^*|^{\kappa-1}$$

It describes how the signal to noise ratio (SNR) varies around the decision boundary  $x^*$ , which in turn determines how easy or hard the classification problem is. For example, if  $\kappa = 1$ , then  $\eta(x)$  jumps by a constant at the decision boundary, making it quite easy to identify. [2] show that one can derive minimax rates for active classification that depend very precisely on  $\kappa$ . A similar idea can be used to describe how a

function varies at the boundary in a level set estimation setting. One can ask a similar question in the convex optimization setting - *Does the behaviour of a particular function around its minimum solely and precisely determine rates for the given function's optimization?*

Having established the clear connection between the notions of UC and TNC, we define the class  $\mathcal{F}_{L,\kappa}^{GUC}(S)$  for any real  $\kappa > 1$  as the class of Generalised  $(L, \kappa)$ -Uniformly Convex or  $(L, \kappa)$ -GUC functions, and we say that  $f \in \mathcal{F}_{L,\kappa}^{GUC}(S)$  if  $f$  is strictly convex, ie,  $f \in \mathcal{F}^C$  and

$$\forall x \in S, \quad f(x) - f(x_f^*) \geq L\|x - x_f^*\|_2^\kappa$$

This can also be thought of as characterizing the SNR of the function near its minimum. Note that  $\kappa$  need not be an integer, and also that a function may lie in several  $\mathcal{F}_{L,\kappa}^{GUC}(S)$  classes. In such a case, what is most relevant is the minimum  $\kappa$  for which the above condition holds, which is unique.

In this paper, we will show how the minimax errors  $\epsilon_T^*(\mathcal{F}_{L,\kappa}^{GUC}), \rho_T^*(\mathcal{F}_{L,\kappa}^{GUC})$  can be cleanly and tightly specified, by providing tight lower and upper bounds in terms of  $T, \kappa$ . We thus argue that the optimal rate of minimizng such a function is determined solely by its behaviour around its optimum.

**Remark 1.** Consider the function  $f(x) = \|x\|_{1.5}^{1.5}$  in  $S = [-2, 2]^d$ .  $f \in \mathcal{F}_\lambda^{SC} \equiv \mathcal{F}_{\lambda,2}^{UC}$  for some  $\lambda$ , because its second derivative is lower bounded on any closed set. This implies that  $f \in \mathcal{F}_{\lambda,2}^{GUC}$ , but since our definition also allows  $1 < \kappa < 2$ , we can also get  $f \in \mathcal{F}_{1,1.5}^{GUC}$ , and this lower  $\kappa = 1.5$  allows us to get faster rates like  $O(T^{-3/2})$  than [4] who will get  $O(T^{-1})$ . (Appendix)

## 4 1-D Stochastic Optimization using Active Learning Algorithms

In this section, we show how to reduce the task of stochastically optimizing a one-dimensional convex function to that of active classification of signs of a monotone gradient. For simplicity of exposition, we assume that the set  $S$  of interest is  $[0, 1]$ , and we have a convex function  $f$  that achieves a unique minimum  $x^*$  inside the set  $(0, 1)$ .

We begin by noting that since  $f$  is convex, its true gradient  $g$  is an increasing function of  $x$ , that is negative to the left of  $x^*$ , 0 at  $x^*$ , and positive to the right of  $x^*$ . Hence, one can think of  $\text{sign}(g(x))$  as being the label of point  $x$ , and finding  $x^*$  corresponds to learning the decision boundary.

In this section, we assume that the oracle returns gradient values corrupted by standard variance gaussian noise. Because this noise is symmetric, when we query at a point to the left of  $x^*$  we are more likely to see a negative (label  $-1$ ), when we are at  $x^*$  we have a 50 – 50 chance of seeing a negative or positive (label 1), and to the right of  $x^*$  we have a greater chance of seeing a positive. So, if we think of  $\eta(x) = P(\text{sign}(g(x) + z) = 1|x)$ , then minimizing the function corresponds to identifying the Bayes classifier  $[x^*, 1]$ . In other words, the point at which  $\eta(x) = 0.5$  is the point at which  $g(x) = 0$ , which is  $x^*$ .

We now argue that an assumption of  $(L, \kappa)$ -GUC for  $f$  implies that for any subgradient  $g_x \in \partial f(x)$ , we have  $\|g_x\|_2 \geq L\|x - x_f^*\|_2^{\kappa-1}$  (Appendix), which then implies a  $(c, \kappa - 1)$ -TNC for  $\eta$ . Here, we shall derive the second implication using the fact that the probability mass of a gaussian random variable  $z$  grows linearly just around its mean (Appendix), which can be stated as

$$\forall t < \sigma, \quad \exists a_1, a_2, \quad a_1 t \leq P(0 \leq z \leq t) \leq a_2 t$$

Let us consider a point  $x$  which is a distance  $t > 0$  to the right of  $x^*$  and hence has label 1 (we can make a similar argument for  $x < x^*$ ). As mentioned above,  $\forall g_x \in \partial f(x)$ ,  $g_x \geq Lt^{\kappa-1}$ . In the presence of gaussian noise, the probability of seeing label 1 is the probability that a draw gets a value in  $(-g_x, \infty)$  so that the sign is not reversed. This yields:

$$\begin{aligned}\eta(x) &= P(g_x + z > 0) = 0.5 + P(-g_x < z < 0) \geq 0.5 + a_1 Lt^{\kappa-1} \\ \implies &\exists c > 0, \quad |\eta(x) - 1/2| \geq c|x - x^*|^{\kappa-1}\end{aligned}$$

[2] analyse an algorithm called the Burnashev-Zigangirov (BZ) algorithm, which is a noise-tolerant variant of binary bisection, under such a TNC. BZ solves the one-dimensional active classification problem such that after making  $T$  queries for a noisy label, it returns a confidence interval  $\hat{I}_T$  which contains  $x^*$  with high probability, and  $\hat{x}_T$  is chosen to be the midpoint of  $\hat{I}_T$ . They provide bounds for the excess risk of  $\int_{[x,1] \Delta [x^*,1]} |2\eta(x) - 1| dx$  where  $\Delta$  is the symmetric difference operator over sets but small modifications to their proofs yield a bound on  $\mathbb{E}|\hat{x}_T - x^*|$ . (Appendix)

The bounded noise setting of  $\kappa = 1$  is easy because the regression function is bounded away from half and we can show an exponential convergence of  $\mathbb{E}(|\hat{x}_T - x^*|) = O(e^{-TL^2/2})$ . The unbounded noise setting of  $\kappa > 1$  is harder because the regression function does not jump and using a variant of BZ analysed in [2], we can show that  $\mathbb{E}(|\hat{x}_T - x^*|) = O\left(\frac{\log T}{T}\right)^{\frac{1}{2\kappa-2}}$  and  $\mathbb{E}(|\hat{x}_T - x^*|^\kappa) = O\left(\frac{\log T}{T}\right)^{\frac{\kappa}{2\kappa-2}}$ . (Appendix)

If  $f$  also obeys a Holder condition with exponent  $\kappa$ ,  $H|x - x^*|^\kappa \geq f(x) - f(x^*) \geq L|x - x^*|^\kappa$ , we can immediately get a bound  $\mathbb{E}[f(x) - f(x^*)] = O\left(\frac{\log T}{T}\right)^{\frac{\kappa}{2\kappa-2}}$ . Interestingly, in the next section on lower bounds, we show that for any dimension,  $\Omega\left(\frac{1}{T}\right)^{\frac{1}{2\kappa-2}}$  is the minimax convergence rate for  $\mathbb{E}(\|\hat{x}_T - x^*\|_2)$  and that  $\Omega\left(\frac{1}{T}\right)^{\frac{\kappa}{2\kappa-2}}$  is the minimax rate for  $\mathbb{E}[f(x) - f(x^*)]$ .

## 5 Optimization Lower Bounds using Active Learning Techniques

Here, we prove lower bounds for  $\epsilon_T^*$ ,  $\rho_T^*$  using an information theory technique that was originally used for proving lower bounds for active classification using the TNC [2], providing a stronger connection between active learning and stochastic convex optimization. Our main result is

**Theorem 1.** *For  $\kappa > 1$ , let  $\mathcal{F}_{L,\kappa}^{GUC}$  be the class of Generalised  $(L, \kappa)$ -Uniformly Convex functions on  $\mathbb{R}^d$ . Then, the minimax rate for function-error and point-error are given by*

$$\epsilon_T^*(\mathcal{F}_{L,\kappa}^{GUC}) = \Omega(T^{-\frac{\kappa}{2\kappa-2}}) \quad , \quad \rho_T^*(\mathcal{F}_{L,\kappa}^{GUC}) = \Omega(T^{-\frac{1}{2\kappa-2}})$$

The proof technique can be summarised as follows. We demonstrate an oracle  $O^*$  and set  $S^*$  over which we prove a lower bound for  $\inf_{\mathcal{M} \in \mathcal{M}_T} \sup_{f \in \mathcal{F}} \mathbb{E}_O[\epsilon(\mathcal{M}, f, S, O)]$ . We go about this by defining a semi-distance between any two elements of our function class as the distance between their minima. We then choose two very similar functions  $f_0, f_1$  whose minima are  $2a$  apart (think of  $a$  as a small constant, getting smaller with increasing  $T$ ). The oracle chooses one of these two functions and the learner gets to query at points  $x$  in domain  $S^*$ , receiving noisy gradient and function values  $y, z$ . We then define distributions corresponding to the two functions  $P_T^0, P_T^1$  and choose  $a$  so that they are at most a constant KL-distance  $\gamma$  apart. We then use a classical Fano's inequality which, using  $a$  and  $\gamma$ , lower bounds the probability of identifying the wrong function by any estimator (and hence optimizing the wrong function), given any finite sample of size  $T$ .

**Theorem 2.** [8] Let  $\mathcal{F}$  be a model class with an associated semi-distance  $\delta(\cdot, \cdot) : \mathcal{F} \times \mathcal{F} \rightarrow \mathbb{R}$  and each  $f \in \mathcal{F}$  having an associated measure  $P^f$  on a common probability space. Let  $f_0, f_1 \in \mathcal{F}$  be such that  $\delta(f_0, f_1) \geq 2a > 0$  and  $KL(P^0 || P^1) \leq \gamma$ . Then,

$$\inf_{\hat{f}} \sup_{f \in \mathcal{F}} P^f \left( \delta(\hat{f}, f) \geq a \right) \geq \inf_{\hat{f}} \max_{j \in \{0,1\}} P^j \left( \delta(\hat{f}, f_j) \geq a \right) \geq \max \left( \frac{\exp(-\gamma)}{4}, \frac{1 - \sqrt{\gamma/2}}{2} \right)$$

For the set of generalised  $(L, \kappa)$ -uniformly convex functions  $\mathcal{F}_{L,\kappa}^{GUC}$ , we choose the set  $S^*$  to be  $[0, 1]^d$ . The chosen oracle  $O^*$  just adds standard normal noise to the true function and gradient values. We first consider a subclass  $\mathcal{U}_{L,\kappa}^{GUC} \subset \mathcal{F}_{L,\kappa}^{GUC}$ , which is chosen such that every point in  $S^*$  is the minimizer of exactly one function in  $\mathcal{U}_{L,\kappa}^{GUC}$  (also,  $f \in \mathcal{U}_{L,\kappa}^{GUC}$  has a unique minimum  $x_f^* \in S^*$ ).

We now bound  $\inf_{\hat{f}_T} \sup_{f \in \mathcal{U}_{L,\kappa}^{GUC}} \mathbb{E} \|\hat{x}_T - x^*\|$ . Define semi-distance  $\delta(f_a, f_b) = \|x_a^* - x_b^*\|$  and

$$\begin{aligned} f_0(x) &= d^{\frac{\kappa}{2}} L \sum_{i=1}^d |x_i|^\kappa = d^{\frac{\kappa}{2}} L \|x\|_\kappa^\kappa, & g_0(x) &= \kappa d^{\frac{\kappa}{2}} L (x_1^{\kappa-1}, \dots, x_d^{\kappa-1}) \\ f_1(x) &= \begin{cases} d^{\frac{\kappa}{2}} L \left( |x_1 - 2a|^\kappa + \sum_{i=2}^d |x_i|^\kappa + (4a)^\kappa - (2a)^\kappa \right) & \text{if } x_1 \in [0, 4a] \\ f_0(x) & \text{otherwise} \end{cases} \\ g_1(x) &= \begin{cases} \kappa d^{\frac{\kappa}{2}} L \left( \frac{|x_1 - 2a|^\kappa}{(x_1 - 2a)}, x_2^{\kappa-1}, \dots, x_d^{\kappa-1} \right) & \text{if } x_1 \in [0, 4a] \\ g_0(x) & \text{otherwise} \end{cases} \end{aligned}$$

for an appropriate value of  $a$ . The minima of these two functions are at  $(0, 0, \dots, 0)$  and  $(2a, 0, \dots, 0)$  respectively, and hence  $\delta(f_0, f_1) = 2a$ . Notice that these two functions and their gradients differ only on a set of size  $4a$ . The functions are both convex and both in  $\mathcal{F}_{L,\kappa}^{GUC}$ . (Appendix)

On querying at point  $x$ , the oracle returns  $z \sim \mathcal{N}(f(x), \sigma^2)$  and  $y \sim \mathcal{N}(g(x), \sigma^2 I_d)$ . So, for  $i \in \{0, 1\}$  we have  $P^i(Z_t, Y_t | X = x_t) = \mathcal{N}((f_i(x_t), g_i(x_t)), \sigma^2 I_{d+1})$ . Let  $X_1^T, Y_1^T, Z_1^T$  be the random variables corresponding to the set of query points and responses. We define the probability distribution corresponding to every  $f \in \mathcal{U}_{L,\kappa}^{GUC}$  to be their joint distribution over  $T$  samples, and so  $P_T^0 := P^0(X_1^T, Y_1^T, Z_1^T)$  and  $P_T^1 := P^1(X_1^T, Y_1^T, Z_1^T)$ .

The KL-divergence of these two distributions can be shown to be  $KL(P_T^0, P_T^1) = O(d^\kappa L^2 T a^{2\kappa-2})$  (proof in Appendix). We choose  $a = (d^\kappa L^2 T)^{-\frac{1}{2\kappa-2}}$  so that  $\exists \gamma > 0$ ,  $KL(P_T^0, P_T^1) \leq \gamma$ .

Since we satisfy the conditions of the theorem, we get  $\inf_{\hat{f}_T} \sup_{f \in \mathcal{U}_{L,\kappa}^{GUC}} P_f(\delta(\hat{f}, f) \geq a) \geq C$  for some constant  $C$ . It immediately follows that

$$\inf_{\hat{f}_T} \sup_{f \in \mathcal{U}_{L,\kappa}^{GUC}} \mathbb{E} \|\hat{x}_T - x_f^*\| \geq a \cdot \inf_{\hat{f}_T} \sup_{f \in \mathcal{U}_{L,\kappa}^{GUC}} P_f(\delta(\hat{f}, f) \geq a) \geq a \cdot C = \Omega \left( (d^\kappa L^2 T)^{-\frac{1}{2\kappa-2}} \right)$$

where the first inequality follows is an application of Markov's inequality, the second follows by the application of the aforementioned Fano's theorem, and the last step follows by the choice of  $a$ . This gives us our required bound on  $\rho_T^*(\mathcal{U}_{L,\kappa}^{GUC})$ , and correspondingly for  $\epsilon_T^*(\mathcal{U}_{L,\kappa}^{GUC})$  because

$$\inf_{\mathcal{M}_T} \sup_{f \in \mathcal{U}_{L,\kappa}^{GUC}} \mathbb{E}[f(\hat{x}_T) - f(x_f^*)] \geq \inf_{\mathcal{M}_T} \sup_{f \in \mathcal{U}_{L,\kappa}^{GUC}} L[\mathbb{E} \|\hat{x}_T - x_f^*\|^\kappa] \geq \inf_{\hat{f}_T} \sup_{f \in \mathcal{U}_{L,\kappa}^{GUC}} L[\mathbb{E} \|\hat{x}_T - x^*\|]^\kappa$$

where the first inequality follows by the TNC, and the second follows by applying Jensen's inequality for  $\kappa > 1$  and because a method  $\mathcal{M}_T$  returning a point  $\hat{x}_T \in S^*$  corresponds exactly to it returning a guessed function  $\hat{f}_T \in \mathcal{U}_{L,\kappa}^{GUC}$  (by construction of  $\mathcal{U}_{L,\kappa}^{GUC}$ ).

Finally, we get the bounds on  $\rho_T^*(\mathcal{F}_{L,\kappa}^{GUC})$  and  $\epsilon_T^*(\mathcal{F}_{L,\kappa}^{GUC})$  because we are now taking sup over the larger class  $\mathcal{F}_{L,\kappa}^{GUC}$ . We carry the dimension dependence around to show that we do not derive a contradiction to [ABRW10], who show that  $\epsilon_T^*(\mathcal{F}^C) = \Omega\left(\sqrt{\frac{d}{T}}\right)$  and  $\epsilon_T^*(\mathcal{F}^{SC}) = \Omega\left(\frac{d}{T}\right)$ .

## 6 Tight Upper Bounds using Generalised Epoch-GD

In this section, we demonstrate an alternate algorithm to the dual-averaging one in [4] that gets the same rates in expectation and with high probability. We consider the Epoch-GD algorithm from [3], that was shown to be optimal when parameterized correctly for strongly convex optimization in expectation and in high probability (upto logarithmic factors), and show that again, when parameterized correctly, the exact same algorithm achieves our required rates, for  $\kappa > 1$ .

We make the assumption that  $f$  is known to be in  $\mathcal{F}_{L,\kappa}^{GUC}$  for some  $\kappa > 1$ ; [3] use  $\kappa = 2$ , while [4] need  $\kappa \geq 2$ . We also assume (like [3],[4]) that the oracle always returns a subgradient estimate  $\hat{g}_x$  such that  $\mathbb{E}\hat{g}_x \in \partial f(x)$  and satisfies  $\|\hat{g}_x\|_2 \leq G$ . This assumption is like a Lipschitz condition that also implies that the subgradient is bounded by  $G$  everywhere. In fact, it implies that the diameter is bounded by  $(GL^{-1})^{\frac{1}{\kappa-1}}$  and the function value is bounded by  $(G^\kappa L^{-1})^{\frac{1}{\kappa-1}}$  (Appendix).

The only difference in our EpochGD algorithm (described in table) from [3] is the update for  $\eta_e$  (we get back their update when  $\kappa = 2$ ). The algorithm runs for  $E = \lfloor \log(\frac{T}{C_0} + 1) \rfloor$  epochs so that the total number of gradient queries is bounded by  $T$ . The following theorems can be shown by mimicing the proof in [3]. Note that, because  $f \in \mathcal{F}_{L,\kappa}^{GUC}$ , we have  $\|\hat{x}_T - x^*\| \leq L^{-1/\kappa}[f(\hat{x}_T) - f(x^*)]^{1/\kappa}$ , giving immediate bounds on the point-error of the final guess  $\hat{x}_T = x_1^{E+1}$ .

**Theorem 3.** *There exists an algorithm  $\text{EpochGD}(S, G, L, \kappa, T)$  for any  $f \in \mathcal{F}_{L,\kappa}^{GUC}$ ,  $\kappa \geq 2$ , that after at most  $T$  gradient queries, returns a point  $\hat{x}_T \in S$ , such that  $\mathbb{E}f(\hat{x}_T) - f(x_f^*) = O(T^{-\frac{\kappa}{2\kappa-2}})$  and by Jensen's,  $\mathbb{E}\|\hat{x}_T - x_f^*\| \leq \left(\mathbb{E}(\|\hat{x}_T - x_f^*\|^\kappa)\right)^{\frac{1}{\kappa}} = O(T^{-\frac{1}{2\kappa-2}})$ .*

The  $\kappa \geq 2$  condition seems to be an artifact of analysis. We think the bound is true for all  $\kappa > 1$ , and we support this claim by demonstrating a high probability bound with no restriction on  $\kappa$ , which immediately implies the bound in expectation as well, for all  $\kappa > 1$ .

To prove a high probability bound, [3] use a different projection operator, that now looks like  $\prod_{S \cap B(x_1^e, r)}$ , meaning they project (using a convex program, say) onto a convex set that is an intersection of the original set and a ball centered at  $x_1^e$  of radius  $r$ . They show how to set  $r, C_0, C_1$  in terms of  $\lambda, G, T$  to get the minimax rate of  $\tilde{O}(1/T)$ . We also use exactly the same projection, and show (Appendix) how to similarly set  $r, C_0, C_1$  in terms of  $L, \kappa, G, T$  to obtain :

**Theorem 4.** *There exists an algorithm  $\text{EpochGDProj}(S, G, L, \kappa, T, \delta)$  for any  $f \in \mathcal{F}_{L,\kappa}^{GUC}$ ,  $\kappa > 1$ , that after at most  $T$  gradient queries, returns  $\hat{x}_T \in S$ , such that  $f(\hat{x}_T) - f(x_f^*) = \tilde{O}(T^{-\frac{\kappa}{2\kappa-2}})$  with probability*

---

**Algorithm 1** EpochGD (set  $S$ , gradient bound  $G$ , time steps  $T$ , GUC parameters  $\kappa, L$ )

---

**Input:** Constants  $C_0, C_1, \kappa$  and total time  $T$ .

Initialize  $x_1^1 \in S$  arbitrarily

Initialize  $T_1 = C_0 \cdot 2$ ,  $\eta_1 = C_1 \cdot 2^{-\frac{\kappa}{2\kappa-2}}$  and  $e = 1$

- 1: **while**  $\sum_{i=1}^e T_i \leq T$  **do**
- 2:   **for**  $t = 1$  to  $T_e$  **do**
- 3:     Query the oracle at  $x_t^e$  to obtain  $\hat{g}_t$
- 4:

$$x_{t+1}^e = \prod_S (x_t^e - \eta_e \hat{g}_t)$$

- 5:   **end for**
- 6:   Set  $x_1^{e+1} = \frac{1}{T_e} \sum_{t=1}^{T_e} x_t^e$
- 7:   Set  $T_{e+1} = 2T_e$ ,  $\eta_{e+1} = \eta_e \cdot 2^{-\frac{\kappa}{2\kappa-2}}$  and  $e \leftarrow e + 1$
- 8: **end while**

**Output:**  $x_1^e$

---

at least  $1 - \delta$  for any  $\delta > 0$ , where  $\tilde{O}$  hides  $\log \log T$  and  $\log(1/\delta)$  factors. Hence,  $\|\hat{x}_T - x_f^*\| = \tilde{O}(T^{-\frac{1}{2\kappa-2}})$  with probability at least  $1 - \delta$ .

The only assumptions are that we know  $f \in \mathcal{F}_{L,\kappa}^{GUC}$  for  $\kappa > 1$ , and that we have a bound on any returned subgradient  $G$ . [4], [3] make the same assumptions, but [4] need  $\kappa \geq 2$ , while [3] assume  $\kappa = 2$ , and hence don't get rates better than  $O(1/T)$  like we do for  $1 < \kappa < 2$ .

## 7 Discussion and Future Work

The most common assumptions in the literature for proving convergence results for optimization algorithms are those of convexity and  $\lambda$ -strong convexity, and [4] prove results for  $(L, \kappa)$ -UC when  $\kappa \geq 2$ . The concept of  $(L, \kappa)$ -GUC is a strictly weaker notion because it is immediately implied by  $(L, \kappa)$ -UC in the realm of  $\kappa \geq 2$ , but has no corresponding notion when  $1 < \kappa < 2$ .  $\kappa \rightarrow \infty$  corresponds to flatter and flatter functions around their minima while  $\kappa \rightarrow 1$  is actually the *best* case with a large SNR (called the bounded noise case in Section 4, as done in [2]) and one can achieve extremely fast rates for this case, that are surprisingly even faster than  $O(1/T)$ .

The lower bound  $\Omega(T^{-\frac{\kappa}{2\kappa-2}})$  for  $\epsilon^*$  that we prove don't contradict those in [1], who show that their method gives the correct dependence on dimension  $d$  as well, but we really wanted to show how the rate decays with  $T, \kappa$ . Setting  $\kappa = 2$  for strongly convex functions, we do recover the well-known  $\Omega(1/T)$  lower bound. Also, letting  $\kappa \rightarrow \infty$ , gives the classic  $\Omega(1/\sqrt{T})$  bound. We also wanted to demonstrate how to use an active learning proof technique, which is novel in its application to optimization, and we believe that it can be modified to give tight rates in  $d$ , with a better construction.

The lower bound  $\Omega(T^{-\frac{1}{2\kappa-2}})$  for  $\rho^*$  is interesting because the optimization literature does not often focus on point-error estimates. We note that these are strongly supported by intuition as we can note by the rate's behaviour at the extremes of  $\kappa$ . If the function has  $\kappa \rightarrow 1$ , it says that we should be able to identify the optimum extremely fast, as supported by our result for the bounded noise setting in 1-D, and also by the



tight upper bounds for  $\rho$  using Epoch-GD. However, when  $\kappa \rightarrow \infty$ , the function starts to look extremely flat around its minimum, and while we can optimize function-error very well (because a wide range of points have function value close to the minimum value), we cannot expect to get close to the true optimum point.

Our upper bounds on  $\epsilon$  and  $\rho$ , when we know that the function is in  $\mathcal{F}_{L,\kappa}^{GUC}$ , involve an appropriately tuned Epoch-Gradient-Descent [3] and the rates match those of [4] who use a dual-averaging algorithm, showing that the lower bounds achieved in terms of  $T, \kappa$  are indeed correct and tight. It is important to note that we make the same kind of assumptions as [4] and [3] - the number of time steps  $T$ , a bound on noisy subgradients  $G$ , a convexity parameter  $L$ , and knowledge of  $\kappa$  ( $\kappa = 2$  for [3], any  $\kappa \geq 2$  for [4], any  $\kappa > 1$  for us). Substituting  $\kappa = 2$  in our algorithm yields their bound of  $O(1/T)$  for strongly convex functions (as well as the same parameter settings as in [3]). Also,  $\kappa \rightarrow \infty$  recovers the classical rate of  $O(1/\sqrt{T})$  for convex functions as well.

In practice, one does not usually know the smoothness of the function at hand, and hence what value of  $\kappa$  to use in the proposed algorithm. Of course, if we only know that the function is convex then we can use any gradient descent algorithm, and if we know that it is strongly convex then we can use  $\kappa = 2$ , so our algorithm is not any weaker than existing ones, but it is certainly stronger if we know  $\kappa$  accurately. [4] have an algorithm in the non-stochastic setting that adapts to unknown  $\kappa$  with the loss of only a  $\log T$  term in the rate. However, in the stochastic setting, it is an open problem to construct an algorithm that can adapt to unknown  $\kappa$ .

Every function has a unique smallest  $\kappa$  that it can possibly satisfy strong convexity with (because there is an inherent flatness to the function at its minimum, and we cannot satisfy *GUC* with a  $\kappa$  smaller than that), and this  $\kappa$  should be learnable with access to noisy function values and gradients. For example, if the function is *simple*, in the sense that it doesn't have different rates of smoothness in different areas, then perhaps we can spend half our budget of  $T$  queries just querying at a point and estimating its gradient in random directions around it to get a good estimate of  $\kappa$ , and then run the algorithm using this estimate. Of course, it is also not clear how the algorithms perform when they use a wrong value for  $\kappa$  (sensitivity of convergence rates to a possible estimation error in  $\kappa$ ).

The lower bound proof proposed here is useful because it bounds  $\epsilon^*$  and  $\rho^*$  simultaneously, by bounding the point-error  $\rho$  and using the GUC condition to bound the function-error  $\epsilon$ . Also, notice that the upper-bound proofs proceed by bounding the function-error  $\epsilon$  and use the GUC condition to bound the point-error  $\rho$ . We conjecture that the same proof should be alterable to get the right dependence on  $d, T, \kappa$  simultaneously, using a larger set of functions (say associated with corners of a hypercube), each function having its optimum perturbed in different dimensions (according to the 1s of its corner). Also, going by the lower bounds of  $\Omega\left(\frac{d}{T}\right)$  for  $\mathcal{F}^{SC}$  and  $\Omega\left(\sqrt{\frac{d}{T}}\right)$  for  $\mathcal{F}^C$ , one might guess that the right dependence on  $d, T, \kappa$  should look like  $\Omega\left(\left(\frac{d}{T}\right)^{\frac{\kappa}{2\kappa-2}}\right)$ .

Our upper bound proofs have a few loose ends. It should be an interesting exercise to get rid of the  $\kappa \geq 2$  condition for the simpler expectation argument, and remove the  $\log \log T$  in the high probability argument. The  $\log \log T$  factor also appears in the analysis by [3], so a tighter analysis in that setting should immediately lend itself to improvements in our bounds. However, this is possibly a secondary concern compared to learning or adapting to  $\kappa$  and getting the right dependence on  $d, \kappa$ .

Hints of connections to active learning have been lingering in the literature, as noted by [6], but as far as we know, nobody has explicitly used concepts, algorithms and proof techniques to connect the two fields

strongly. It is interesting to note, however, that while many active learning methods degrade exponentially with dimension  $d$ , the rates in optimization degrade polynomially. This may limit the use of algorithms from active learning, which are possibly trying to solve harder problems, like learning a  $d - 1$ -dimensional decision boundary or level set, while optimization problems in any dimension are really interested in getting to a single good point. However, we feel that this is just the start of stronger conceptual ties between the two fields.

## 8 Acknowledgments

Thanks a ton to Rob Nowak, Alekh Agarwal and Sivaraman Balakrishnan for valuable insights! This research is supported in part by AFOSR under grant FA9550-10-1-0382 and NSF under grant IIS-1116458.

## References

- [1] A. Agarwal, P.L. Bartlett, P. Ravikumar, and M.J. Wainwright. Information-theoretic lower bounds on the oracle complexity of stochastic convex optimization. *Information Theory, IEEE Transactions on*, (99):1–1, 2010.
- [2] R.M. Castro and R.D. Nowak. Minimax bounds for active learning. In *Proceedings of the 20th annual conference on learning theory*, pages 5–19. Springer-Verlag, 2007.
- [3] E. Hazan and S. Kale. Beyond the regret minimization barrier: an optimal algorithm for stochastic strongly-convex optimization. COLT, 2011.
- [4] A. Iouditski and Y. Nesterov. Primal-dual subgradient methods for minimizing uniformly convex functions. *Universite Joseph Fourier, Grenoble, France [Report]*, 2010.
- [5] A.S. Nemirovsky and D.B. Yudin. *Problem complexity and method efficiency in optimization*. John Wiley & Sons, 1983.
- [6] M. Raginsky and A. Rakhlin. Information complexity of black-box convex optimization: A new look via feedback information theory. In *Communication, Control, and Computing, 2009. Allerton 2009. 47th Annual Allerton Conference on*, pages 803–510. IEEE, 2009.
- [7] A.B. Tsybakov. Optimal aggregation of classifiers in statistical learning. *The Annals of Statistics*, 32(1):135–166, 2004.
- [8] A.B. Tsybakov. *Introduction to Nonparametric Estimation*. Springer Series in Statistics. Springer, 2009.

## Appendix (References)

[CN07] Castro & Nowak, (2007) Minimax Bounds for Active Learning. *COLT 2007*

[HK11] Hazan & Kale (2011) Beyond The Regret Minimization Barrier: An Optimal Algorithm for Stochastic Strongly-Convex Optimization. *COLT 2011*

## SECTION 3

We now justify the claim that no function (including  $f(x) = \|x\|_{1.5}^{1.5} = \sum_i |x_i|^{1.5}$ ) can satisfy Uniform Convexity for  $\kappa < 2$ , but they can satisfy Generalized Uniform Convexity for  $\kappa < 2$ .

If uniform convexity could be satisfied for (say)  $\kappa = 1.5$ , then  $\forall x, y \in S$

$$f(y) - f(x) - g_x^\top(y - x) \geq \frac{\lambda}{2} \|x - y\|_2^{1.5}$$

Take  $x, y$  both on the positive  $\mathbf{x}$ -axis. The Taylor expansion would require, for some  $c \in [x, y]$ ,

$$f(y) - f(x) - g_x^\top(y - x) = (x - y)^\top H(c)(x - y) \leq \|H(c)\|_F \|x - y\|_2^2$$

Now, taking  $\|x - y\|_2 = \epsilon \rightarrow 0$  by choosing  $x$  closer to  $y$ , the Taylor condition requires the residual to grow like  $\epsilon^2$  (going to zero fast), but the UC condition requires the residual to grow at least as fast as  $\epsilon^{1.5}$  (going to zero slow). At some small enough value of  $\epsilon$ , this would not be possible. Since the definition of UC needs to hold for all  $x, y \in S$ , this gives us a contradiction. So,  $f \notin \mathcal{F}_{L,1.5}^{UC}$ .

However, one can note that  $x_f^* = 0$ , and  $f(x) - f(x_f^*) = \|x\|_{1.5}^{1.5} = \|x - x_f^*\|_{1.5}^{1.5}$ , hence  $f \in \mathcal{F}_{1,1.5}^{GUC}$ .

## SECTION 4

This section deals with reducing 1D convex optimization to active learning of gradient signs.

**Lemma 1.** *If  $f \in \mathcal{F}_{L,\kappa}^{GUC}$ , then for any subgradient  $g_x \in \partial f(x)$ , we have  $\|g_x\|_2 \geq L\|x - x^*\|_2^{\kappa-1}$ .*

*Proof.* By convexity, we have

$$f(x^*) \geq f(x) + g_x^\top(x^* - x)$$

Rearranging terms and since  $f \in \mathcal{F}_L^\kappa$ , we get

$$g_x^\top(x - x^*) \geq f(x) - f(x^*) \geq L\|x - x^*\|_2^\kappa$$

By Holder's inequality,

$$\|g_x\|_2 \|x - x^*\|_2 \geq g_x^\top(x - x^*)$$

Putting them together, we have

$$\|g_x\|_2 \|x - x^*\|_2 \geq L\|x - x^*\|_2^\kappa$$

giving us our result. □

**Lemma 2.** *For a gaussian random variable  $z$ ,  $\forall t < \sigma$ ,  $\exists a_1, a_2$ ,  $a_1 t \leq P(0 \leq z \leq t) \leq a_2 t$*

*Proof.* We wish to characterize how the probability mass of a gaussian random variable grows just around its mean. Our claim is that it grows linearly with the distance from the mean, and the following simple argument argues this neatly.

Consider a  $X \sim N(0, \sigma^2)$  random variable at a distance  $t$  from the mean 0. We want to bound  $\int_{-t}^t d\mu(X)$  for very small  $t$ . The key idea in bounding this integral is to approximate it by a smaller and larger rectangle, each of the rectangles having a width  $2t$  (from  $-t$  to  $t$ ).

The first one has a height equal to  $\frac{e^{-t^2/2\sigma^2}}{\sigma\sqrt{2\pi}}$ , the smallest value taken by the gaussian in  $[-t, t]$  achieved at  $t$ , and the other with a height equal to the  $\frac{1}{\sigma\sqrt{2\pi}}$ , the largest value of the gaussian in  $[-t, t]$  achieved at 1.

The smaller rectangle has an area of  $2t \frac{e^{-t^2/2\sigma^2}}{\sigma\sqrt{2\pi}} \geq 2t \frac{e^{-1/2}}{\sigma\sqrt{2\pi}}$  when  $t < \sigma$ . The larger rectangle clearly has an area of  $2t \frac{1}{\sigma\sqrt{2\pi}}$ .

Hence we have  $A_1 t = 2t \frac{1}{\sigma\sqrt{2\pi}e} \leq P(|X| < t) \leq 2t \frac{1}{\sigma\sqrt{2\pi}} = A_2 t$  for  $t < \sigma$ . Similarly, for a one-sided inequality, we have  $a_1 t = t \frac{1}{\sigma\sqrt{2\pi}e} \leq P(0 < X < t) \leq t \frac{1}{\sigma\sqrt{2\pi}} = a_2 t$  for  $t < \sigma$ .

We note that the gaussian tail inequality  $P(X > t) \leq \frac{1}{t} e^{-t^2/2\sigma^2}$  really makes sense for large  $t > \sigma$  and we are interested in  $t < \sigma$ . There are tighter inequalities, but for our purpose, this will suffice.  $\square$

We now move to proving the key results claimed in the section.

**Lemma 3.** *If  $|\eta(x) - 1/2| \geq L$ , the midpoint  $\hat{x}_T$  of the high-probability interval returned by BZ satisfies  $\mathbb{E}|\hat{x}_T - x^*| = O(e^{-TL^2/2})$ .*

*Proof.* This is a subpart of a proof from [CN07], who note that the BZ algorithm works by dividing  $[0, 1]$  into a grid of  $m$  points (interval size  $1/m$ ) and makes  $T$  queries (only at gridpoints) to return an interval  $\hat{I}_T$  such that  $\Pr(x^* \notin \hat{I}_T) \leq m e^{-TL^2}$ . We choose  $\hat{x}_T$  to be the midpoint of this interval, and hence get

$$\begin{aligned} \mathbb{E}|\hat{x}_T - x^*| &= \int_0^1 \Pr(|\hat{x}_T - x^*| > u) du \\ &= \int_0^{1/2m} \Pr(|\hat{x}_T - x^*| > u) du + \int_{1/2m}^1 \Pr(|\hat{x}_T - x^*| > u) du \\ &\leq \frac{1}{2m} + \left(1 - \frac{1}{2m}\right) \Pr\left(|\hat{x}_T - x^*| > \frac{1}{2m}\right) \\ &\leq \frac{1}{2m} + m e^{-TL^2} = O\left(e^{-TL^2/2}\right) \end{aligned}$$

for the choice of the number of gridpoints as  $m = e^{TL^2/2}$ .  $\square$

**Lemma 4.** *If  $|\eta(x) - 1/2| \geq L|x - x^*|^\kappa$ , the point  $\hat{x}_T$  obtained from a modified version of BZ satisfies  $\mathbb{E}|\hat{x}_T - x^*| = O\left(\left(\frac{\log T}{T}\right)^{\frac{1}{2\kappa-2}}\right)$  and  $\mathbb{E}[|\hat{x}_T - x^*|^\kappa] = O\left(\left(\frac{\log T}{T}\right)^{\frac{\kappa}{2\kappa-2}}\right)$ .*

*Proof.* We again follow the same proof as in [CN07]. Initially, they assume that the grid points are not aligned with  $x^*$ , ie  $\forall k \in \{0, \dots, m\}$ ,  $|x^* - k/m| \geq 1/3m$ . This implies that for all gridpoints  $x$ ,  $|\eta(x) -$

$1/2 \geq L(1/3m)^{\kappa-1}$ . Following the exact same proof above, with this new  $L$ ,

$$\begin{aligned}
\mathbb{E}[|\hat{x}_T - x^*|^\kappa] &= \int_0^1 \Pr(|\hat{x}_T - x^*|^\kappa > u) du \\
&= \int_0^{(1/2m)^\kappa} \Pr(|\hat{x}_T - x^*| > u^{1/\kappa}) du + \int_{(1/2m)^\kappa}^1 \Pr(|\hat{x}_T - x^*| > u^{1/\kappa}) du \\
&\leq \left(\frac{1}{2m}\right)^\kappa + \left(1 - \left(\frac{1}{2m}\right)^\kappa\right) \Pr\left(|\hat{x}_T - x^*| > \frac{1}{2m}\right) \\
&\leq \left(\frac{1}{2m}\right)^\kappa + m \exp(-TL^2(1/3m)^{2\kappa-2}) = O\left(\left(\frac{T}{\log T}\right)^{\frac{1}{2\kappa-2}}\right)
\end{aligned}$$

on choosing  $m$  proportional to  $\left(\frac{T}{\log T}\right)^{\frac{1}{2\kappa-2}}$ .

In [CN07], they elaborate in detail how to get rid of the assumption that the grid points don't align with  $x^*$ . They use a more complicated variant of BZ with three interlocked grids, and gets the same rate as above without that assumption. The reader is directed to their exposition for clarification.  $\square$

## SECTION 5

**Lemma 5.**  $d^{\frac{\kappa}{2}} L \sum_{i=1}^d |x_i|^\kappa =: f_0(x) \in \mathcal{F}_{L,\kappa}^{GUC}$ , for all  $\kappa > 1$

*Proof.* Firstly, this is the sum of convex functions and is hence convex. Also,  $f_0(x_{f_0}^*) = 0$  at  $x_{f_0}^* = 0$ . So, all we need to show is that  $f_0(x) - f_0(x_{f_0}^*) \geq \|x - x_{f_0}^*\|_2^\kappa$ , or in other words

$$d^{\frac{\kappa}{2}} L \sum_{i=1}^d |x_i|^\kappa \geq L \|x\|_2^\kappa \Leftrightarrow \left(\sum_{i=1}^d |x_i|^\kappa\right)^{1/\kappa} \geq \frac{1}{\sqrt{d}} \|x\|_2$$

which is true since if  $\kappa \leq 2$ , we know  $\|x\|_\kappa \geq \|x\|_2$ , and as  $\kappa \rightarrow \infty$ ,  $\|x\|_\infty \geq \frac{1}{\sqrt{d}} \|x\|_2$ .

$f_1$  is continuous at  $x_1 = 4a$  (in fact the constants were chosen that way). The gradient at  $x_1 = 4a$  increases from  $\kappa d^{\frac{\kappa}{2}} L (2a)^{\kappa-1}$  to  $\kappa d^{\frac{\kappa}{2}} L (4a)^{\kappa-1}$ . Hence convexity is preserved at the kink. Since both parts of  $f_1$  are convex, and convexity is maintained at the kink, we conclude that  $f_1$  is convex.

Now, look at  $f_0(x)$  for  $x_1 \leq 4a$ . It is actually just  $f_0(x)$ , but translated by  $2a$  in direction  $x_1$ , with a constant added, and hence has the same GUC parameters. Now, the part with  $x_1 > 4a$  is just  $f_0(x)$  itself, which have the same GUC parameters as the part with  $x_1 \leq 4a$ . So  $f_1(x) \in \mathcal{F}_{L,\kappa}^{GUC}$  also.  $\square$

Now we bound the KL divergence of the two probability distributions  $P^i(z_t, y_t | X = x_t) = \mathcal{N}((f_i(x_t), g_i(x_t), \sigma^2 I_{d+1}))$  and  $P_T^i := P^i(X_1^T, Y_1^T, Z_1^T)$  (for  $i = 0, 1$ ).

**Lemma 6.** For  $P_T^0, P_T^1$  as defined in terms of  $f^0, f^1$ ,  $\text{KL}(P_T^0, P_T^1) = O(Ta^{2\kappa-2})$

$$\begin{aligned}
\text{KL}(P_T^0, P_T^1) &= \mathbb{E}^0 \left[ \log \frac{P^0(X_1^T, Y_1^T, Z_1^T)}{P^1(X_1^T, Y_1^T, Z_1^T)} \right] \\
&= \mathbb{E}^0 \left[ \log \frac{\prod_{t=1}^T P^0(Y_t, Z_t | X_t) P(X_t | X_1^{t-1}, Y_1^{t-1}, Z_1^{t-1})}{\prod_{t=1}^T P^1(Y_t, Z_t | X_t) P(X_t | X_1^{t-1}, Y_1^{t-1}, Z_1^{t-1})} \right] \tag{3}
\end{aligned}$$

$$\begin{aligned}
&= \mathbb{E}^0 \left[ \log \frac{\prod_{t=1}^T P^0(Y_t, Z_t | X_t)}{\prod_{t=1}^T P^1(Y_t, Z_t | X_t)} \right] \\
&= \sum_{t=1}^T \mathbb{E}^0 \left[ \mathbb{E}^0 \left[ \log \frac{P^0(Y_t, Z_t | X_t)}{P^1(Y_t, Z_t | X_t)} \middle| X_1, \dots, X_T \right] \right] \\
&\leq T \max_{x \in [0,1]^d} \mathbb{E}^0 \left[ \log \frac{P^0(Y_1, Z_1 | X_1)}{P^1(Y_1, Z_1 | X_1)} \middle| X_1 = x \right] \\
&= T \max_{x \in [0,1]^d} \mathbb{E}^0 \left[ \log \frac{P^0(Y_1 | X_1) P^0(Z_1 | X_1)}{P^1(Y_1 | X_1) P^1(Z_1 | X_1)} \middle| X_1 = x \right] \tag{4}
\end{aligned}$$

$$\begin{aligned}
&\leq T \left( \max_{x \in [0,1]^d} \mathbb{E}^0 \left[ \log \frac{P^0(Y_1 | X_1)}{P^1(Y_1 | X_1)} \middle| X_1 = x \right] + \max_{x \in [0,1]^d} \mathbb{E}^0 \left[ \log \frac{P^0(Z_1 | X_1)}{P^1(Z_1 | X_1)} \middle| X_1 = x \right] \right) \\
&= \frac{T}{2} \left( \max_{x \in [0,1]^d} \|g_0(x) - g_1(x)\|_2^2 + \max_{x \in [0,1]^d} (f_0(x) - f_1(x))^2 \right) \tag{5}
\end{aligned}$$

$$= \frac{T d^\kappa L^2}{2} \left( \kappa^2 \max_{x_1 \in [0,4a]} \left( \frac{|x_1 - 2a|^\kappa}{(x_1 - 2a)} - x_1^{\kappa-1} \right)^2 + \max_{x_1 \in [0,4a]} (|x_1 - 2a|^\kappa - x_1^\kappa)^2 \right) \tag{6}$$

$$= O(d^\kappa L^2 T a^{2\kappa-2}) + O(d^\kappa L^2 T a^{2\kappa}) = O(d^\kappa L^2 T a^{2\kappa-2}) \tag{7}$$

(3) follows because the distribution of  $X_t$  conditional on  $X_1^{t-1}, Y_1^{t-1}, Z_1^{t-1}$  depends only on the algorithm  $\mathcal{M}_T$  and does not change with the underlying distribution. (4) follows because conditioned on  $X_t$ ,  $Y_t \perp Z_t$ . We also used  $(Y_i, Z_i | X_i) \perp (Y_j, Z_j | X_j)$  for  $i \neq j$ . (5) follows because the KL-divergence between two identity-covariance gaussians is just half the squared euclidean distance between their means. (6) follows by simply substituting the gradient/function values and because the functions/gradients differ only on  $x_1 \in [0, 4a]$ . (7) follows by checking values at  $0, 2a, 4a$ , and the smaller power is larger order since  $a < 1$ , treating  $\kappa$  as a constant.

## SECTION 6

We begin by showing that the assumption of  $f$  having a bounded subgradient on  $S$  corresponds to assuming a bound on the diameter of  $S$ , and hence on the maximum achievable function value.

**Lemma 7.** *If  $f \in \mathcal{F}_{L,\kappa}^{GUC}$  with  $\|\nabla f(x)\|_2 \leq G$ , then  $\forall x \in S$ , we have  $\|x - x_f^*\|_2 \leq (GL^{-1})^{\frac{1}{\kappa-1}} =: D$  (diameter) and also that  $f(x) - f(x_f^*) \leq (G^\kappa L^{-1})^{\frac{1}{\kappa-1}} =: M$  (maximum)*

*Proof.* This follows the corresponding proof in [HK11]. For any  $x \in S$ , let  $g_x \in \partial f(x)$ . By convexity,  $f(x_f^*) \geq f(x) + g_x^\top (x_f^* - x)$ , and so  $f(x) - f(x_f^*) \leq g_x^\top (x - x_f^*) \leq \|g_x\|_2 \|x - x_f^*\|_2$  (by Holder's

inequality), implying that  $G\|x - x_f^*\|_2 \geq f(x) - f(x_f^*) \geq L\|x - x_f^*\|_2^\kappa$ .

From this we get  $\|x - x_f^*\|_2^{\kappa-1} \leq G/L$  or  $\|x - x_f^*\|_2 \leq G^{\frac{1}{\kappa-1}}/L^{\frac{1}{\kappa-1}}$ . Finally  $f(x) - f(x_f^*) \leq G\|x - x_f^*\|_2 \leq G^{\frac{\kappa}{\kappa-1}}/L^{\frac{1}{\kappa-1}}$ . Note that for strongly convex functions,  $\kappa = 2$ , and [HK11] observe that  $f(x) - f(x_f^*) \leq G^2/L$  and  $\|x - x_f^*\|_2 \leq G/L$ .  $\square$

**Lemma 8.** [HK11] Applying  $T$  iterations of the update  $x_{t+1} = \prod_S(x_t - \eta \hat{g}_t)$ , where  $\hat{g}_t$  is an unbiased estimator for a subgradient  $g_t$  of convex function  $f$  at  $x_t$  that satisfies  $\|\hat{g}_t\| \leq G$ , we get the following bound for  $\bar{x} = \frac{1}{T} \sum_t x_t$

$$\mathbb{E}f(\bar{x}) - f(x_f^*) \leq \frac{\eta G^2}{2} + \frac{\|x_1 - x_f^*\|^2}{2\eta T}$$

Define  $\Delta_e = f(x_1^e) - f(x_f^*)$ . The corresponding proof in [HK11] shows  $\mathbb{E}\Delta_e \leq 2G^2\eta_e$ . We point out that the bound for  $\Delta_{E+1}$  with  $E = \lfloor \log(\frac{T}{C_0} + 1) \rfloor$  yields Theorem 3 immediately.

**Lemma 9.** For any  $e$ , with  $T_e = C_0 2^e$ ,  $\eta_e = C_1 \cdot 2^{-e\frac{\kappa}{2\kappa-2}}$ , for appropriate  $C_0, C_1, C_2$ , we have

$$\mathbb{E}\Delta_e \leq C_2 \eta_e$$

*Proof.* We choose  $C_0 = 1$  and prove the lemma by performing induction on  $e$ .

The first step of the induction, for  $e = 1$ , requires  $\Delta_1 \leq C_2 \eta_1 = C_2 C_1 2^{-\frac{\kappa}{2\kappa-2}}$ . **[R1]**

Assume the hypothesis is true until  $e$ , and we prove it for  $e+1$ . Let  $\mathbb{E}_e$  denote the expectation conditioned on the randomness until epoch  $e$ . By Lemma 8,

$$\mathbb{E}_e[\Delta_{e+1}] \leq \frac{\eta_e G^2}{2} + \frac{\|x_1^e - x^*\|^2}{2\eta_e T_e} \leq \frac{\eta_e G^2}{2} + \frac{\Delta_e^{2/\kappa}}{2\eta_e T_e L^{2/\kappa}}$$

where the second inequality follows because  $\Delta_e \geq L\|x_1^e - x^*\|^\kappa$ .

Now taking expectation for all epochs upto  $e$ , and applying Jensen's for  $\kappa \geq 2$ , we get

$$\mathbb{E}[\Delta_{e+1}] \leq \frac{\eta_e G^2}{2} + \frac{[\mathbb{E}\Delta_e]^{2/\kappa}}{2\eta_e T_e L^{2/\kappa}} \leq \frac{\eta_e G^2}{2} + \frac{C_2^{2/\kappa} \eta_e^{2/\kappa}}{2\eta_e T_e L^{2/\kappa}}$$

where the second inequality follows by the induction hypothesis).

Now, we would like the second term  $\frac{C_2^{2/\kappa} \eta_e^{2/\kappa}}{2\eta_e T_e L^{2/\kappa}} \leq \frac{\eta_e G^2}{2}$ , so that their sum is  $\leq \eta_e G^2$ . **[R2]**

We now want the sum of the two terms  $\eta_e G^2 \leq C_2 \eta_{e+1}$ , so that the induction can go through. **[R3]**

We will now show values for  $C_1$  and  $C_2$  for which all 3 conditions hold (we chose  $C_0 = 1$ ).

From **[R3]**, we derive  $C_2 \geq G^2 \frac{\eta_e}{\eta_{e+1}} = G^2 2^{\frac{\kappa}{2\kappa-2}}$ , giving a lower bound for  $C_2$ .

From **[R2]**, we derive  $\eta_e^{\frac{2\kappa-2}{\kappa}} \geq \frac{C_2^{2/\kappa} 2^{-e}}{G^2 L^{2/\kappa}} \Leftrightarrow \eta_e \geq \frac{C_2^{\frac{1}{\kappa-1}}}{G^{\frac{k}{\kappa-1}} L^{\frac{1}{\kappa-1}}} \times 2^{-e\frac{\kappa}{2\kappa-2}}$ . Since  $\eta_e = C_1 \cdot 2^{-e\frac{\kappa}{2\kappa-2}}$ ,

we get  $C_1 \geq \frac{C_2^{\frac{1}{\kappa-1}}}{G^{\frac{k}{\kappa-1}} L^{\frac{1}{\kappa-1}}} \geq \frac{G^{\frac{2}{\kappa-1}} 2^{\frac{\kappa}{2(\kappa-1)^2}}}{G^{\frac{k}{\kappa-1}} L^{\frac{1}{\kappa-1}}} = \frac{G^{\frac{2-\kappa}{\kappa-1}} 2^{\frac{\kappa}{2(\kappa-1)^2}}}{L^{\frac{1}{\kappa-1}}}$ , giving a lower bound for  $C_1$  using  $C_2$ .

For **[R1]** to hold, note that its right hand side is  $C_2\eta_1 = C_2C_12^{-\frac{\kappa}{2\kappa-2}} \geq G^22^{\frac{\kappa}{2\kappa-2}} \frac{G^{\frac{2-\kappa}{\kappa-1}} 2^{\frac{\kappa}{2(\kappa-1)^2}}}{L^{\frac{1}{\kappa-1}}} 2^{-\frac{\kappa}{2\kappa-2}} = \frac{G^{\frac{\kappa}{\kappa-1}}}{L^{\frac{1}{\kappa-1}}} \cdot 2^{\frac{\kappa}{2(\kappa-1)^2}} = M \cdot 2^{\frac{\kappa}{2(\kappa-1)^2}}$ . Hence, **[R1]** requires us to show that  $\Delta_1$  is smaller than the RHS, which is at least as large as  $M \cdot 2^{\frac{\kappa}{2(\kappa-1)^2}}$ . Since  $\kappa > 1$ , this is trivially true, since we already know that  $\Delta_1 \leq M$  (Lemma 7).

Hence, we have proved the lemma for the constants  $C_0 = 1, C_1 = \frac{G^{\frac{2-\kappa}{\kappa-1}} 2^{\frac{\kappa}{2(\kappa-1)^2}}}{L^{\frac{1}{\kappa-1}}}, C_2 = G^22^{\frac{\kappa}{2\kappa-2}}$ . For the result in [HK11],  $C_0 = 1, C_1 = 2/L, C_2 = 2G^2$  works, which we get with  $\kappa = 2$ .  $\square$

**Lemma 10.** [HK11] Let  $R$  be an upper bound on  $\|x_1 - x_f^*\|$ . Applying  $T$  iterations of the update  $x_{t+1} = \Pi_{S \cap B(x_1^e, R)}(x_t - \eta \hat{g}_t)$ , where  $\hat{g}_t$  is an unbiased estimator for the subgradient of convex  $f$  at  $x_t$  satisfying  $\|\hat{g}_t\| \leq G$ . For  $\bar{x} = \frac{1}{T} \sum_t x_t$  and any  $\delta > 0$ , with probability at least  $1 - \delta$ , we have

$$f(\bar{x}) - f(x_f^*) \leq \frac{\eta G^2}{2} + \frac{\|x_1 - x_f^*\|^2}{2\eta T} + \frac{4GR\sqrt{2\log(1/\delta)}}{\sqrt{T}}$$

Define  $\Delta_e = f(x_1^e) - f(x_f^*)$ . The corresponding proof in [HK11] shows  $\Delta_e \leq 6G^2\eta_e$  with probability at least  $(1 - \frac{\delta}{E})^{e-1}$ . We point out that the bound for  $\Delta_{E+1}$  with  $E = \lfloor \log(\frac{T}{C_0} + 1) \rfloor$  yields Theorem 4 immediately by noting that  $(1 - \frac{\delta}{E})^E \geq 1 - \delta$ .

**Lemma 11.** For any epoch  $e$  and any  $\delta > 0$ ,  $T_e = C_02^e$ ,  $E = \lfloor \log(\frac{T}{C_0} + 1) \rfloor$ ,  $\eta_e = C_12^{-e\frac{\kappa}{2\kappa-2}}$ , for appropriate choice of  $C_0, C_1, C_2$ , we have with probability at least  $(1 - \frac{\delta}{E})^{e-1}$

$$\Delta_e \leq C_2\eta_e$$

*Proof.* We let  $\tilde{\delta} = \frac{\delta}{E}$  and prove the lemma by induction on  $e$ .

The first step of induction,  $e = 1$ , requires  $\Delta_1 \leq C_2\eta_1 = C_2C_12^{-\frac{\kappa}{2\kappa-2}}$ . **[R1]**

Assume that  $\Delta_e \leq C_2\eta_e$  for some  $e \geq 1$ , with probability at least  $(1 - \tilde{\delta})^{e-1}$  and we now prove it for epoch  $e + 1$ . We condition on the event  $\Delta_e \leq C_2\eta_e$  which happens with the above probability. By the GUC,  $\Delta_e \geq L\|x_1^e - x^*\|^\kappa$ , and the conditioning implies that  $\|x_1^e - x^*\| \leq (C_2\eta_e/L)^{1/\kappa}$ , which we choose as the radius  $r$  of the ball for the EpochGDProj projection.

Lemma 10 applies with  $R = (\frac{C_2\eta_e}{L})^{\frac{1}{\kappa}}$  and so with probability at least  $1 - \tilde{\delta}$ , we have

$$\Delta_{e+1} \leq \frac{\eta_e G^2}{2} + \frac{\|x_1^e - x^*\|^2}{2\eta_e T_e} + \frac{4G(\frac{C_2\eta_e}{L})^{\frac{1}{\kappa}} \sqrt{2\log(\frac{1}{\tilde{\delta}})}}{\sqrt{T_e}} \leq \frac{\eta_e G^2}{2} + \frac{C_2^{\frac{2}{\kappa}} \eta_e^{\frac{2}{\kappa}}}{2\eta_e T_e L^{\frac{2}{\kappa}}} + \frac{4G(\frac{C_2\eta_e}{L})^{\frac{1}{\kappa}} \sqrt{2\log(\frac{1}{\tilde{\delta}})}}{\sqrt{T_e}}$$

where the second inequality again follows because  $\|x_1^e - x^*\| \leq (C_2\eta_e/L)^{1/\kappa}$ .

We would now like the second term  $\frac{C_2^{\frac{2}{\kappa}} \eta_e^{\frac{2}{\kappa}}}{2\eta_e T_e L^{\frac{2}{\kappa}}} \leq \frac{\eta_e G^2}{6}$  **[R2]** and also the third term  $\frac{4G(\frac{C_2\eta_e}{L})^{\frac{1}{\kappa}} \sqrt{2\log(\frac{1}{\tilde{\delta}})}}{\sqrt{T_e}} \leq \frac{\eta_e G^2}{3}$  **[R3]**, so the sum of all three terms would be  $\leq \eta_e G^2$ .

Lastly, we would like  $\eta_e G^2 \leq C_2\eta_{e+1}$  **[R4]** so that the induction goes through.



Then, factoring in the conditioned event which happens with probability at least  $(1 - \tilde{\delta})^{e-1}$  we would get  $\Delta_{e+1} \leq C_2 \eta_{e+1}$  with probability at least  $(1 - \tilde{\delta})^e$ .

Now, we show values for  $C_0, C_1, C_2$  such that the four conditions hold.

For **[R4]**, we need  $\eta_e G^2 \leq C_2 \eta_{e+1}$ , and hence we get  $C_2 \geq G^2 2^{\frac{\kappa}{2\kappa-2}}$ , a lower bound for  $C_2$ .

For **[R2]**, we need  $\frac{\eta_e G^2}{6} \geq \frac{C_2^{\frac{2}{\kappa}} \eta_e^{\frac{2}{\kappa}}}{2\eta_e T_e L^{\frac{2}{\kappa}}} \Leftrightarrow \eta_e^{\frac{2\kappa-2}{\kappa}} \geq \frac{3C_2^{\frac{2}{\kappa}}}{G^2 L^{\frac{2}{\kappa}} C_0} 2^{-e}$  from which we get that  $\eta_e \geq \left(\frac{3}{G^2 C_0}\right)^{\frac{\kappa}{2\kappa-2}} \left(\frac{C_2}{L}\right)^{\frac{1}{\kappa-1}} 2^{-e \frac{\kappa}{2\kappa-2}}$ , giving  $C_1 \geq \left(\frac{3}{G^2 C_0}\right)^{\frac{\kappa}{2\kappa-2}} \left(\frac{C_2}{L}\right)^{\frac{1}{\kappa-1}}$  since  $\eta_e = C_1 2^{-e \frac{\kappa}{2\kappa-2}}$ .

For **[R3]**, we need  $\frac{\eta_e G^2}{3} \geq \frac{4G \left(\frac{C_2 \eta_e}{L}\right)^{\frac{1}{\kappa}} \sqrt{2 \log(1/\tilde{\delta})}}{\sqrt{T_e}} \Leftrightarrow \eta_e^{\frac{\kappa-1}{\kappa}} \geq \frac{12C_2^{1/\kappa} \sqrt{2 \log(1/\tilde{\delta})}}{GL^{1/\kappa} C_0^{1/2} 2^{e/2}}$ , which yields  $C_1 \geq \left(\frac{3(96 \log(1/\tilde{\delta}))}{G^2 C_0}\right)^{\frac{\kappa}{2\kappa-2}} \left(\frac{C_2}{L}\right)^{\frac{1}{\kappa-1}}$  since  $\eta_e = C_1 2^{-e \frac{\kappa}{2\kappa-2}}$ . This is the stronger condition on  $C_1$ .

For **[R1]** to hold, we note that its right hand side is  $C_2 \eta_1 = C_1 C_2 2^{-\frac{\kappa}{2\kappa-2}} \geq G^2 2^{\frac{\kappa}{2\kappa-2}} \left(\frac{3(96 \log(1/\tilde{\delta}))}{G^2 C_0}\right)^{\frac{\kappa}{2\kappa-2}} \left(\frac{C_2}{L}\right)^{\frac{1}{\kappa-1}} 2^{-\frac{\kappa}{2\kappa-2}} = \frac{G^{\frac{\kappa}{\kappa-1}}}{L^{\frac{1}{\kappa-1}}} \left(\frac{288 \log(1/\tilde{\delta})}{C_0}\right)^{\frac{\kappa}{2\kappa-2}} 2^{\frac{\kappa}{2(\kappa-1)^2}} = M 2^{\frac{\kappa}{2(\kappa-1)^2}}$  if  $C_0 = 288 \log(1/\tilde{\delta})$ . So, **[R1]** requires that we need that  $\Delta_1$  to be smaller than the RHS which is larger than  $M 2^{\frac{\kappa}{2(\kappa-1)^2}}$ , which is trivially true since  $\Delta_1 \leq M$  (Lemma 7) and  $\kappa \geq 1$ .

Hence, we have proved the lemma for  $C_0 = 288 \log(E/\delta)$ ,  $C_1 = \frac{G^{\frac{2-\kappa}{\kappa-1}} 2^{\frac{\kappa}{2(\kappa-1)^2}}}{L^{\frac{1}{\kappa-1}}}$ ,  $C_2 = G^2 2^{\frac{\kappa}{2\kappa-2}}$ .

[HK11] use  $C_0 = 288 \log(E/\delta)$ ,  $C_1 = 2/L$ ,  $C_2 = 2G^2$  works, which we get with  $\kappa = 2$ .

As with [HK11], because of the changed  $T_1$ , they lose a factor of  $\log \log T$ , because the total number of epochs is now smaller. Another way of seeing this, like in [HK11], is to allow the total number of epochs to be  $E = \lfloor \log(\frac{T}{288} + 1) \rfloor$  instead of  $E = \lfloor \log(\frac{T}{C_0} + 1) \rfloor = \lfloor \log(\frac{T}{288 \log(E/\delta)} + 1) \rfloor$ . Then, the algorithm runs for  $T \log \log T$  steps, to give a bound of  $O(1/T)$  with high probability. We can easily reverse this to show that the algorithm runs for  $\tilde{T}$  steps, to give a bound of  $O(\log \log \tilde{T}/\tilde{T})$  with high probability.  $\square$